Published in final edited form as:
J Proteome Res. 2009 May ; 8(5): 2241–2252. doi:10.1021/pr800678b.

A Ranking-Based Scoring Function For Peptide-Spectrum Matches

Ari M. Frank[*]
Department of Computer Science and Engineering, University of California, San Diego. 9500 Gilman Drive, Mail Code 0404 La Jolla, CA 92093-0404, USA

Abstract

The analysis of the large volume of tandem mass spectrometry (MS/MS) proteomics data that is generated these days relies on automated algorithms that identify peptides from their mass spectra. An essential component of these algorithms is the scoring function used to evaluate the quality of peptide-spectrum matches (PSMs). In this paper, we present new approach to scoring of PSMs. We argue that since this problem is at its core a ranking task (especially in the case of de novo sequencing), it can be solved effectively using machine learning ranking algorithms. We developed a new discriminative boosting-based approach to scoring. Our scoring models draw upon a large set of diverse feature functions that measure different qualities of PSMs. Our method improves the performance of our de novo sequencing algorithm beyond the current state-of-the-art, and also greatly enhances the performance of database search programs. Furthermore, by increasing the efficiency of tag filtration and improving the sensitivity of PSM scoring, we make it practical to perform large-scale MS/MS analysis, such as proteogenomic search of a six-frame translation of the human genome (in which we achieve a reduction of the running time by a factor of 15 and a 60% increase in the number of identified peptides, compared to the InsPecT database search tool). Our scoring function is incorporated into PepNovo+ which is available for download or can be run online at: http://bix.ucsd.edu.

Keywords

MS/MS; scoring; peptide; PSM; de novo; database search; machine learning; ranking; boosting

Introduction

In the post-genomic era, focus has shifted towards the identification and characterization of all gene products that are expressed in a given organism.[1] This large-scale analysis of proteins, dubbed *Proteomics*, contributes greatly to the understanding of gene function, biochemistry of proteins, processes and pathways.[2] In this arena, tandem mass spectrometry (MS/MS) has emerged as the tool of choice for high-throughput identification of proteins and determination of details of their primary structures.[3,4] Recent technological breakthroughs have led to a dramatic increase in the volume of MS/MS data being generated. However, analyzing this abundant data is not a trivial task. Without developing novel, more powerful algorithms, we can expect computational bottlenecks to restrict the scope of discoveries that can be made from experiments involving mass spectrometry.

[*] E-mail: E-mail: arf@cs.ucsd.edu.

Supporting Information Available This information is available free of charge via the Internet at http://pubs.acs.org.

MS/MS-based proteomics typically involves the enzymatic digestion of a protein sample, followed by separation and fragmentation of peptide ions in the mass spectrometer. The initial computational task performed when analyzing this data involves the identification of peptides from their mass spectra. While some algorithms perform this task by comparing query spectra to collections of identified spectra (e.g., spectrum libraries[5,6]) or theoretical spectra derived from a database of peptide sequences (such as the SEQUEST algorithm[7]), most algorithms rely on scoring functions to assess the quality of matches between a query spectrum and its possible peptide interpretations. A peptide-spectrum match (PSM) scoring function assigns a numerical value to a peptide-spectrum pair (P,S) expressing the likelihood that the fragmentation of a peptide with sequence P is recorded in the experimental mass spectrum S. The problem of scoring PSMs has received considerable interest in the community over the years.[8-25] Most scoring functions create a *generative* statistical model for $Prob(S|P)$, the probability that spectrum S was created from a fragmentation of peptide P. We can then chose $P^* = \text{argmax}_P Prob(S|P)$ as the likeliest peptide that created S. The hope is, that if the model $Prob(S|P)$ is sufficiently accurate, the peptide that maximizes $Prob(S|P)$ will indeed be the true peptide whose fragmentation is recorded in S. The probability $Prob(S|P)$ is often converted to a score by using a log ratio test that compares $Prob(S|P)$ to the probability obtained from a simple null hypothesis, such as a model that assumes that all peaks are distributed randomly.[8,13,16,17,19,22,23]

The problem with using generative approaches to scoring PSMs is that peptide fragmentation is an extremely complex process, that is not easily represented with high fidelity by simple statistical models. Current scoring functions are usually sufficient when the search space is small, such as searching against a small set of protein sequences. However, when we increase the search space, by searching against a large database such as the six-frame translation of the human genome, or by performing de novo sequencing (which effectively searches the space of all peptides), the number good identifications typically decreases significantly. In these large search spaces, generative scoring models often lack the sufficient power to discriminate between the correct PSMs and the many close false ones (such as homeometric peptides that often exhibit very similar fragmentation patterns[26]).

It is worth mentioning that scoring functions for PSMs are used slightly differently with database searches and de novo sequencing. In de novo sequencing, we search the space of all peptides, and we assume that the correct peptide sequence is in our search space (with this statement we ignore the fact that a peptide might contain modifications since it is often too difficult to perform de novo sequencing in such cases without knowledge of the peptide's specific modification). Therefore the PSM scoring function has a simple *ranking* goal in this case; with each spectrum, bring the correct peptide to the top of the PSM candidate list. In contrast, when we score a PSM from a database search, we cannot assume that the correct peptide is necessarily in the database. In fact, in many cases, the database does not contain the correct peptide sequence for a query spectrum (e.g., the peptide might originate from an unknown gene or span an unknown splice boundary). Thus, a database search PSM scoring function has a more difficult ranking goal. In addition to bringing the correct PSM to top of the spectrum's candidate list, the scoring function also needs to (ideally) assign *all* correct PSMs a higher score than *all* incorrect PSMs. From a machine learning perspective, this can be viewed as *classification* task, in which we create a model that is trained to separate between the class of all correct PSMs and the class of incorrect ones. This task can be quite difficult because often PSM scores vary according to the quality of the spectrum.[27] Thus, a correct match between a poorly fragmented peptide and spectrum A might score lower than an incorrect match between some other well fragmented peptide and spectrum B. Nonetheless, since their search space is typically small, database scoring functions usually cope with these more stringent requirements (they are rarely confronted with too many high-scoring incorrect PSMs). In addition, database scores are often post-processed,[28,29] where additional factors are considered, such as the difference between the score of the top and second best PSMs (e.g., the ΔC_n measure in SEQUEST's score[7]). Thus, in many cases, low-scoring PSMs are often accepted because the gap between the score of the first and second best PSMs is large enough. However, when the size of the database search space increases substantially, such as when searching a six-frame translation of a genome, there are many more strong but incorrect PSMs, and the database scoring function's performance deteriorates significantly. As we demonstrate below, a more powerful and discriminatory scoring function becomes essential in these circumstances.

The difference in the goals of scoring PSMs from de novo sequencing and database search has led us to choose a less conventional machine learning algorithm to train models. Instead of the more common generative probabilistic models, we take a discriminative ranking approach to scoring with the boosting-based RankBoost algorithm.[30] Though we are not the first too use discriminative models for scoring PSMs (e.g., LOD scores[15]), this is typically not a common approach. By using discriminative algorithms, we do not attempt to train models that describe the process of peptide fragmentation (like the generative approaches), but rather, we optimize the models directly on the PSM scoring task at hand: distinguishing between correct PSMs and incorrect ones.

To train effective models we rely on a diverse set of features that capture various aspects of the quality of a PSM (e.g., the number of annotated peaks, the score of the peptide's path in a spectrum graph, etc.). We also use features that examine how well the observed spectrum fits predictions we make for the peptide's fragment peak ranks, using a novel ranking-based algorithm we developed.[31] Though many of the features we use in our models are, each on their own, only slightly helpful at indicating if a PSM is correct or not, the RankBoost algorithm combines them into a powerful discriminatory scoring function for PSMs.

Much of the success of our algorithm can be attributed to the large volume of MS/MS data that is now available for training. We collected a set of ~320000 examples of unique peptide-spectrum pairs. With such a large dataset, the learning algorithm was able to create detailed models using hundreds of features without a fear of detrimental overfitting of the training data. While originally designed with de novo sequencing in mind, we show how our scoring approach can be effectively applied to database search results too.

Our method is effective for several PSM scoring tasks. When we used it to rerank the output of our de novo sequencing algorithm PepNovo,[16] it significantly improved the performance over a strong de novo algorithm like Peaks.[12] We also applied the score to the problem of peptide sequence tag generation, and were able to generate more accurate and longer tags, which improved the performance of a database search (making InsPecT[17] runs approximately 15 times faster). We created specific scoring models for rescoring database search results, training them to behave more like a classifiers and not just pure rankers of PSMs. When we applied these models to post-process the output of InsPecT, it significantly increased the number of peptide identifications (a 20% increase when searching against a 30 million amino acid database). The improvement was much more significant when we applied our score to the results of a search against a six-frame translation of the human genome (~3 billion amino acids), where the number of identified peptides increased by 60%. This combination of both a significantly faster, and at the same time, much more sensitive database search, makes it practical to perform large scale proteogenomic analysis, even with large eukaryotic genomes.

Methods

MS/MS Datasets

Our experiments with scoring used a large set of ~ 320000 unique peptide-spectrums pairs collected from various MS/MS experiments using low-resolution CID ion-trap mass spectrometers. Most of the data we used was generated in the Briggs lab at UCSD (samples of from human HEK293 cell culture[32] and samples from *Dictyostelium discoideum*[33]), and the Smith lab at PNNL (samples taken from *Shewanella oneidensis* MR-1[34,35]).

We relied on the InsPecT database search tool to perform peptide identifications (release 20070613),[17] using the default search parameters (precursor mass tolerance 2.5 Da, fragment ion tolerance 0.5 Da). All searches were performed using a shuffled decoy database.[36,37] The InsPecT *F*-score threshold values for accepting identifications were selected to ensure a true positive peptide identification rate of 98% (i.e., only 2% of the peptide hits came from the decoy database). Since a peptide's charge and precursor mass can greatly influence the nature of its observed spectrum, we partitioned the training data into distinct sets as described in Table 1, and trained separate models for each partition.

Ranking Algorithm

Machine learning deals with algorithms that enable a computer to “learn” from data (inductive learning). A common machine learning task is classification, where a learning algorithm is used to derive a model for assigning a class label to each instance x from a domain space of instances $\mathscr{X}$ (e.g., the Perceptron algorithm,[38] Naïve Bayesian classifiers,[39] Support Vector Machines[40]). However, classification algorithms are not always the most suitable framework; some problems have an inherent structure that suggests using other frameworks. For instance, a query to an internet search engine may return many webpages as answers. Usually, one cannot state that an answer to a query is completely right or completely wrong, rather a common approach is to assign a degree of relevance to each returned webpage. In such cases we can use a *ranking* algorithm which scores the answers on a gradient presenting the most relevant answers first (when the ranking algorithm is used to refine a previous ordering it is also called *reranking*). As we see below, ranking is also useful approach to solving problems involving PSM scoring.

We used the RankBoost algorithm of Freund et al.[30] to train our ranking-based models. The RankBoost algorithm uses a machine learning method called boosting,[41,42] which produces highly accurate prediction rules by combining many “weak” rules that, each on their own, may be only moderately accurate.

To train models with RankBoost we need to supply the algorithm with several inputs:

- $\mathscr{X}$ - a instance space. In our case, $\mathscr{X}$ is a training set of PSMs, in which each instance $x \in \mathscr{X}$ is a pair $x = (P,S)$, where P is a peptide sequence and S is a mass spectrum.
- Φ - an ordering of the instances in the training set. This information is described using a feedback function $\Phi : \mathscr{X} \times \mathscr{X} \rightarrow \mathbb{R}$ that contains ordered pairs of instances of the form (x_0,x_1). In our case, we used a simple feedback function that assign 1 to a pair (x_0,x_1) if both x_0 and x_1 were PSMs belonging to the same spectrum S and x_1 has the correct peptide P, while x_0 is a PSM with some other incorrect peptide P'. All other pairs of instances were given a weight of 0.
- F - a set of feature functions. Each feature $f \in F$ can be expressed as a function $f : \mathscr{X} \rightarrow \mathbb{R}$. Our features measure different aspects of the quality of the PSM (such as the number of observed fragment ions, the offset of the precursor match, etc.) We go into further details about the feature functions below.

The goal of the RankBoost learning algorithm is to create a scoring function $H : \mathscr{X} \to \mathbb{R}$ that weights and combines feature functions from F, in order to induce a ranking that misorders as few pairs as possible in Φ. In our case, this means that the ranking function H is optimized to give correct PSMs a higher score than incorrect ones (when confronted with pairs of PSMs that involve the same spectrum S).

One of the benefits of the RankBoost algorithm is that it only incorporates into its models features that are useful (i.e., they help it make fewer ordering mistakes on the training data). RankBoost also has a high tolerance to uninformative features; we can supply it with a large pool of candidate features that can be correlated, and some might not even be relevant to the objective we are trying to achieve, nevertheless, in the training process, the algorithm performs its own internal “feature selection”, and incorporates only features that advance the goal.[30] This property makes it easy for us to design a single set of feature functions (“one size fits all”) that incorporates all features that *might* be useful for peptide identification, without needing to consider whether the features are important for a specific model. For example, a feature that looks at the number of y^{+2} annotations might be very important for a model for scoring large triply-charged peptides, but only represent noise in a model that scores singly-charged peptides. With RankBoost we do not have to evaluate each model and decide which of the possible features are relevant to it, the algorithm does that automatically for us.

Feature functions for scoring Peptide-Spectrum Matches

The most important component in our scoring models are the feature functions they use. Our models draw on a diverse set of features, created using domain knowledge, that each in their own way reflect different characteristics that can help distinguish between correct and incorrect PSMs. In total, the models can contain up to 225 features, though as explained above, not all get selected in each model. We grouped these features into different classes, as described below. A more detailed description of the feature functions can be found in the online supplemental material.

Peak Rank Prediction Features

There are many chemical pathways that participate in peptide fragmentation and lead to the generation of spectra with very diverse characteristics. Better understanding of peptide fragmentation process is needed for more accurate and sensitive peptide sequencing algorithms.[43,44] One way in which knowledge of peptide fragmentation “rules” is used to improve PSM scoring is by predicting theoretical spectra for candidate peptides, and comparing them to the observed experimental spectrum (e.g., checking that the two spectra share the same strong peaks, or high value for the dot-product between them). We note that unlike the method described in the SEQUEST algorithm,[7] the theoretical spectra predicted with the aid of the additional chemical knowledge do not treat all *b*- and *y*-ions equally, but rather try to determine which specific peaks should be strongest (e.g., peaks that are *N*-terminal to proline). This gives more discriminatory power since the peaks in the experimental spectrum must not only appear in the correct mass locations, but they also need to have the correct “shape” (i.e., specific peaks need to have a stronger intensity). If the models governing the generation of the theoretical spectra are accurate, then there should be a high resemblance between the theoretical and observed spectra. Using this approach has led to improved peptide identification rates in database searches.[24,45]

Since predicting accurate MS/MS spectra is a difficult task,[46] we chose to solve a slightly simpler problem, predicting a ranks of a peptide's fragment ions. This problem is formally described in Figure 1. For more details on our algorithm to solve this problem see ref.[31]

Once we have a prediction for the ranks of a peptide's fragment ions, we compare them with the ranks observed for the fragments in the experimental spectrum using several simple feature functions. For instance, such a function might report the observed rank (in the experimental spectrum) of a fragment predicted to be strongest according to our model, others may report the observed ranks of the fragments predicted to be second-strongest, third-strongest, etc. (the features focus mostly on the strongest peaks since those are predicted most accurately). We also created features that report if there are noticeable gaps in our rank predictions. For example, a feature can report how many of the peaks that are predicted to be strong, are actually missing in the experimental spectrum.

Spectrum Graph Features

The space of all peptides is extremely large, making it inappropriate for an exhaustive case-by-case analysis. Nonetheless, most de novo algorithms are able to consider all likely peptides by modeling the search space for a query spectrum as a spectrum graph.[8,47] A *spectrum graph* is a directed acyclic graph; its vertices correspond to putative prefix masses (cleavage sites) of the peptide. Two vertices are connected by a directed edge from the vertex with the lower mass to the one with a higher mass if the difference between them equals the mass of an amino acid. We use PepNovo's scoring function to score nodes in the graph.[16] For each sequence peptide P we can assign a path in the spectrum graph; the score of the path is indicative of how likely it is that the observed spectrum was created from the fragmentation of P.

The spectrum graph feature functions examine several aspects that measure the quality of P's path in the graph. They report the path score, average path score (normalized according to the peptide length), the lowest score of a node in the graph (which can be indicative of a sequencing error), the rank of path in the graph, etc.

Peak annotation features

The spectrum graph scores evaluate combinations of fragments that involve specific cleavage sites. It is also beneficial to take a global look at how well the peptide explains the spectrum's peaks, like in the case of the aforementioned peak rank prediction features. With the peak annotation features, we look at more basic statistics that examine the quality of PSMs using functions that simply count a peptide's matched peaks. For example, these features look at the number of annotated peaks amongst the strongest 25 and 50 peaks in the experimental spectrum. Others report the proportion of explained intensity in the experimental spectrum, or count the number of *b*-,*y*-, y^{+2}-ions, etc., that are observed in the given PSM.

Peak offset features

When annotating fragment ions, we generally tolerate a mass differences of up to 0.5 Da between the expected mass of a fragment, as computed from the peptide sequence, and the actual mass observed in the spectrum. However, most of the true fragment peaks observed in spectra are much closer to their expected mass, usually being less than 0.1 Da away. A peptide that has many fragment peaks with a relatively large offset from their expected mass is likely to be relying on spurious opportunistic peak matches, and is therefore more likely to be incorrect. This type of peak offset information is most useful with the most abundant fragment ions, which are *b*,*y*, so offset related features focus only on them. Such feature will look at attributes like the average mass offset of all identified *b*- or *y*-ion peaks and the maximal mass offset for *b*- or *y*-peaks from their expected position.

Sequence Composition Features

Proteins are not random sequences of amino acids. They often contain conserved, or characteristic patterns that are responsible for inducing a specific spacial conformation or for providing certain function. In addition, certain amino acid patterns are more likely to be ionized and detected using MS/MS than others (e.g., basic amino acids are usually required for effective peptide ionization). These observations gave rise to the notion of proteotypic peptides,[48-50] peptides that are most likely to be confidently identified by MS/MS methods. Many of the characteristics of proteotypic peptides can be captured using simple features that pertain to the peptide's amino acid composition (see supplemental information for more details).

Experimental Results

We now turn to examine how our new scoring model can be used to improve the results of de novo sequencing and database searches. The supplemental material has additional experimental results, describing additional database search results with a standard protein mixture and experiments with peptide sequence tag generation. We first describe the process involved in training PSM scoring models for de novo sequencing.

Model Training For Reranking de Novo Sequences

In order to train our models, we used the partitioning of the training data into 13 sets according to charge and precursor mass, as described in Table 1, and trained a separate PSM scoring model for each partition. Each partition of the training data contained 13000-45000 spectra which were used to create about 400000 training samples as described below. We ran PepNovo on each of training spectrum S_i, and retained the 2000 top-scoring sequences that were at least 6 amino acids long. If none of the 2000 de novo sequences was correct, we excluded the spectrum from the training set. Otherwise, we extracted the highest scoring correct peptide sequence and used it as the positive sample P_i^+ (the highest scoring correct sequence was usually also the longest correct sequence). From the remaining ~2000 incorrect de novo sequences we randomly sampled k = 10-30 sequences and used them as negative samples $P_i^{-1},\ldots,P_i^{-k}$. Note that we did not sample the sequences uniformly from ranks 1-2000, rather we gave more weight to higher ranking sequences which typically are responsible for most of the ranking errors.

Using the feature functions described above, we created an instance x_i^+ in the feature space to represent the PSM of P_i^+ and S_i, and similarly created instances $x_i^{-1},\ldots,x_i^{-k}$ to represent the PSMs of the incorrect de novo sequences. The intances x_i^+ and $x_i^{-1},\ldots,x_i^{-k}$ were used to create a set of k ordered pairs $\left\{\left(x_i^{-j},x_i^+\right)|j=1,\ldots,k\right\}$, which were added to the model's feedback function Φ (with all pairs in Φ having the same weight). We randomly selected 70% of the spectra's sets for training, while the remaining 30% served as a validation set which was used to determine when to terminate a model's training (to avoid overfitting).

Training each score model usually required less than 100 CPU hours, typically converging after less than 100000 training rounds. Figure 2 depicts the progression of the training of the model for doubly charged peptides with precursor masses 1100-1300 Da. The left side of the figure shows the training and validation errors (the error rate represents the proportion of the training PSM pairs in which an incorrect PSM scores higher than a correct one). Most of the ranking error is eliminated early on, falling from 45% to 10% within the first 50 rounds. The figure also shows that most of the error reduction is done using a small set of features; the graph on the right of the figure shows that approximately 25 features are required to achieve the aforementioned error reduction. Continuing to 10000 rounds lowers the validation error to 5.35%, which is only 0.22% higher than lowest validation error 5.13%, that is obtained after 90000 rounds. Therefore, by not seeking to fully optimize the model, we can save considerable time with the training. The graph on the left also shows that as the training progresses overfitting starts to become a problem (this is evident from the widening gap between the training error and validation error that does not decrease at the same pace). However, despite the overfitting, the overall validation error kept on decreasing, and we ended up choosing the model configuration that had the lowest validation error.

De Novo Sequencing Benchmark Results

De novo sequencing of low-resolution MS/MS data is a difficult task. It is unreasonable to expect high accuracy rates from single de novo predictions, since often there are many similarly high-scoring candidates to choose from. Furthermore, some of the commonly used applications for de novo sequencing, such as database filtration with peptide sequence tags[17,51-55] or homology-based database searches,[56-58] actually perform better when supplied with multiple de novo predictions. It is in these circumstances that the advantage of ranking comes into play. Not only does our score increase the accuracy of the top predicted sequence, but it also significantly increases the chances of having a correct sequence in a small set of candidates.

We conducted several benchmark experiments to test the performance of our new scoring function in the context of de novo sequencing. We used several test datasets, including two test sets that were previously used in the literature:

- OPD280 - A set of 280 spectra of doubly charged spectra used to benchmark PepNovo and other de novo programs.[16,18,23]
- ISB769 - A set of 769 spectra from the ISB dataset,[59] which was used in ref.[23]
- HEK8, HEK10, HEK12 - 3 Sets of 1000 doubly charged spectra that were selected from the HEK293 dataset[32]. Each set contained spectra of peptides of specific lengths: 8,10, and 12 amino acids, respectively.

Previous de novo sequencing benchmark experiments mostly focused on predicting a single sequence per spectrum.[16,18,23,60] In these cases it made sense to look at the precision (ratio of correct amino acids in the predictions). However, when predicting multiple sequences, this notion is not well defined. Instead, we examine the proportion of test spectra for which one of the de novo predictions is completely correct, and also look at the rank in which a correct prediction first occurs.

Most publicly available de novo sequencing algorithms typically return a single sequence prediction. The newly developed MS-Dictionary algorithm[61] takes a novel approach of combining de novo sequences and a database search. It uses dynamic programming and probabilistic scoring to generate large ranked lists (dictionaries) of possible peptides for query spectra. These peptides are then compared to a database via pattern matching. We examined MS-Dictionary's results with two settings, one that assumes that the peptides are tryptic (and thus lists only peptides that end with *K* or *R*), and the other that makes no assumption about which digestion enzyme was used. In addition, we ran experiments with the Peaks[12] de novo sequencing algorithm (Peaks Online 2.0) which is one of the best commercial de novo sequencing algorithms available (in preliminary experiments it performed better than other programs such as NovoHMM[18] and MSNovo[23]). The Peaks de novo algorithm only outputs 5 sequences per query spectrum.

Our experiments proceeded as follows. For each spectrum tested, we ran PepNovo and generated the top 2000 scoring sequences. We then reranked the sequences using the our novel scoring function. In the results described below we compare between the algorithm's performance with and without the reranking stage. On average the running time required per spectrum was 1-2 seconds, depending on the peptide's length. This running time usually divided equally between the de novo sequencing and the reranking. When the true peptide sequence was short (8-10 amino acids long), PepNovo typically predicted a complete sequence. However, with longer peptides, whose spectra are often incomplete and lacking detected fragment ion peaks for the amino acids near the terminals, PepNovo sometimes only predicted partial sequences (akin to long sequence tags).

Figure 3 shows benchmark results of the algorithms on the OPD280 and ISB769 datasets. In both datasets we see that PepNovo has significantly higher rates of correct predictions, especially when we look at a small set of de novo solutions. PepNovo uses a much more sophisticated scoring scheme than MS-Dictionary, which explains the large performance gap when small sets of predictions are concerned. In addition, MS-Dictionary is designed to predict only complete de novo sequences. Both the OPD280 and ISB769 datasets include some sequences longer than 14 amino acids (the length limit for which MS-Dictionary is deemed effective), which also explains the lower performance of this algorithm on these datasets. The Peaks algorithm displays accuracy rates that are slightly lower than PepNovo's without ranking. In Figure 3 we also see that reranking de novo sequences significantly increases the accuracy rates. There is an increase of 15-20% when considering small sets of predictions (1-10 sequences). The performance gap is still very significant for 50 and 100 predicted sequences, were the ranked PepNovo results practically reach the maximum they can attain (which is the value for the regular PepNovo results at 2000 sequences). With sets larger than 100 sequences, the gap naturally narrows until the two curves meet at 2000. These results show that the reranking is capable of taking correct, but poorly scoring sequences, and pushing them ahead in ranks. Often the sequences that get pushed forward are shorter than the top-scoring sequences returned by PepNovo. This phenomenon is especially common with spectra of peptides that have poor fragmentation. In such cases, the spectrum graph contains only a partial subpath that corresponds to the correct peptide, and this path frequently gets elongated with spurious edges. PepNovo ends up outputting these incorrect high-scoring sequences ahead of the lower-scoring correct ones. Our new ranking score detects many of these incidents and rectifies the ranks accordingly. This also explains why the average top reranked sequence tends to be shorter than the average top-ranked PepNovo sequence (see Table 2 for the average prediction lengths of the different algorithms).

Table 2 might suggest that PepNovo's superior performance could be attributed to its prediction of shorter incomplete sequences. To rule out this possibility, we benchmarked the algorithms on the task of predicting complete de novo sequences. In these experiments we discarded any prediction that did not span the entire mass range (this applies only to PepNovo and Peaks, since MS-Dictionary always predicts complete peptides). Note, that this puts PepNovo in a slight disadvantage compared to MS-Dictionary, since PepNovo's spectrum graph is not likely to contain a complete path for poorly fragmented peptides, while MS-Dictionary's search space includes all possible peptides.

We used several test sets taken form the HEK293 dataset[32] in which all spectra belong to peptides with specific lengths: 8,10, and 12 amino acids. Figure 4 depicts the results of these experiments. For each peptide length, PepNovo's results are much more accurate than MS-Dictionary's (with as much as 30% more correct sequences for small sets of predicted sequences). Only when very large prediction sets are examined (200 sequences with length 8 and 500 sequences for length 10), does MS-Dictionary catch up with PepNovo's performance. These additional identifications made by MS-Dictionary belong to poorly fragmented peptides that do not have complete paths in PepNovo's spectrum graph. It is likely that PepNovo would be able to predict correct partial sequences in these cases. PepNovo's performance without ranking is at par with Peaks (Peaks has slightly better performance for length 8, while PepNovo has better performance with lengths 10 and 12). However, when PepNovo's results are reranked using our new scoring function, PepNovo exhibits a significant performance boost. The accuracy of the top predicted sequence rises by 10%-15%, and this gap is maintained even when we examine sets of 5, which is the maximal number of sequences generated by Peaks for each query spectrum.

Finally we note that PepNovo's superior performance at de novo sequencing can be harnessed to produce more accurate (and longer) peptide sequence tags for database filtration. PepNovo-generated tags can be 100 times more efficient than the ones used by InsPecT, which can lead to a 15-fold reduction in database search time (see results below). The supplemental material to this manuscript describes experiments we conducted that demonstrate PepNovo's improved tag generation capabilities.

Rescoring Database Search Results

We now turn to experiments that demonstrate how our ranking-based scoring function improves the performance of database searches. Training these models was done slightly differently than the training of the models for reranking de novo results. Instead of using incorrect de novo predictions for false PSMs, we used incorrect database search results (obtained from a run against a large set of shuffled protein sequences). This was done because the search space in a database search is much smaller than the space of all peptides, and thus generates “weaker” incorrect PSMs. In addition, we selected the training pairs of PSMs a bit differently, to make the ranking score become more classification oriented. Instead of having 100% of the pairs of PSMs come from the same spectrum (as was the case with de novo), we found that for database scoring, optimal results were obtained when only 20% of the pairs were selected this way. For the remaining 80% we used pairs of PSMs from different spectra (i.e., we added instances to the model's feedback function that ranked a correct PSM of spectrum S ahead of an incorrect PSM of spectrum S'). This way, we gave a higher weight to the classification-oriented goal of an ideal database scoring function, which aims to bring correct PSMs ahead of the incorrect PSMs from all other spectra, as opposed to the ranking-oriented goal of a de novo scoring function, that is just required to bring the correct PSM ahead of the incorrect PSMs from the same spectrum.

In the experiments described below we used the InsPecT database search tool[17] in a variety of ways (see details below). This search engine is known perform much faster than standard commercial tools like SEQUEST (using a single CPU), and often produces a larger number of peptide identifications. The supplemental material to this manuscript contains experimental results that demonstrate the InsPecT identifies between 6% to 32% more peptides when searching ISB's standard protein mixtures.[62] Our experimental results below demonstrate how our ranked-based scoring function can significantly improve upon InsPecT's search results.

Benchmark Experiments With Human Protein Sequences

To benchmark the performance of our new scoring method on standard MS/MS datasets, we chose an independent run from the HEK293 dataset,[32] consisting of ~750000 spectra. We used InsPecT to perform the database search against the IPI human protein sequence database (version 3.42, containing ~30M amino acids). The searches were conducted in three different modes:

- Regular InsPecT - The default mode for running InsPecT (relies on InsPecT's tagging and scoring functions).
- InsPecT Tags + Rank Score - We take the regular output from InsPecT (which supplies 10 candidate peptides per spectrum), and rescores the PSMs using our ranking-based scoring function.
- PepNovo Tags + Rank Score - We supply InsPecT with sets of PepNovo-generated peptide sequence tags for database filtration. InsPecT finds the top ten scoring peptides for each spectrum (using InsPecT's scoring function). Following that, we post-process the results and rescore the PSMs using our ranking-based scoring function.

The final post-processing step of the database search, in all three running modes, was to filter the results to maintain a false discovery rate of 1% at the spectrum level, which corresponds to approximately 4% at the peptide level (i.e., 1% of the reported spectrum identifications and 4% of the reported peptide identifications are expected to be false positives).

The tags generated by PepNovo (used only in the “PepNovo Tags + Rank Score” mode) were a mixture of 3 tags of length 4, 35 tags of length 5 and 100 tags of length 6. This mixture of tags is approximately 100 times more efficient than the tags used by InsPecT's regular search that uses 25 tags of length 3 (see supplemental material). Since there are many fixed-time operations involved with the database search (file I/O, scanning the database, etc.), there is not a direct linear relationship between the tagging efficiency and the actual run-time speedup. Thus, PepNovo's 100 times more efficient tags led to an approximately 15-fold reduction in InsPecT's run-time.

Table 3 reports the results of these benchmark experiments. The table shows the number of spectra and peptides identified in each of the three search modes, and breaks down the results according to charge states. The total number of identified peptides is lower than the sum of the identifications made with charges 1-3 because often the same peptides were identified through several spectra with different charge states, and we only reported the number of unique peptide identifications. The maximal number of identifications was obtained using the method “PepNovo Tags + Rank Score” (18.8% more spectra and 22.9% more peptides than the regular InsPecT run). The largest increase in identifications was seen with charge 1, where the number of identified spectra rose by 90.5% and the number of peptides rose by 76.1% higher, compared to the number of identifications obtained with InsPecT. This indicates that InsPecT's scoring models do a poorer job with singly-charged peptides, compared to their handling of doubly-charged ones. The results for “InsPecT tags + Rank Score” also show a considerable improvement compared to the default InsPecT run, with an increase of 13.7% in the number of identified spectra and 14.7% in the number of identified peptides. When we consider these figures along with the improvement of +18.8% spectra and +22.9% peptides obtained with “PepNovo Tags + Rank Score”, we can conclude that almost 2/3 of the improvement of “PepNovo Tags + Rank Score” can be attributed to our improved scoring, while the rest of the gained identifications come from PepNovo's more accurate tags. Note that PepNovo's tags yield more identifications than InsPecT's tags despite the fact that they are 100 times more efficient. Interestingly, for triply-charged peptides, the results with “InsPecT Tags + Rank Score” are better than the results obtained with PepNovo's tags. This means that for triply-charged spectra InsPecT's tags are more accurate than PepNovo's. The reason for this is that triply-charged peptides typically have poor fragmentation, so in many cases it is quite difficult to extract long tags (4,5 or 6 amino acids long), while still relatively easy to get a good tag that is only 3 amino acid long.

Benchmark Experiments With Six-Frame Translation Of Human Genome

Despite advances in genome sequencing and gene annotation algorithms, many genes remain unidentified even in the well-studied organisms.[63,64] Annotation of genes using evidence of protein expression obtained via MS/MS experiments (“proteogenomic mapping”) is suggested as a complementary method to sequence-based gene prediction algorithms.[32,35,65-73] Since proteogenomic studies involve searching mass spectra against all possible reading frames in a genome (a “six-frame translation”), the process can be quite time consuming when large eukaryotic genomes are investigated. In addition, the large search space encountered in proteogenomic studies leads also to lower sensitivity (fewer identifications) compared to searches against smaller protein databases.[65,74]

There have been several recent proteogenomic studies involving the six-frame translations of the human genome.[67,69,71] However, these studies used relatively slow search programs such as SEQUEST and X!-Tandem,[75] or relied on high-resolution FTMS to reduce the number of candidates that need to be considered.[71] In our experiments, searching a single spectrum against a six-frame translation of the human genome required approximately 5 minutes of CPU time using InsPecT, which was benchmarked as being 10 times faster than X!-Tandem and 60 times faster than SEQUEST on a single processor.[73] The recently developed MS-Dictionary algorithm[61] is capable of performing rapid searches, on the order of a second per spectrum, against large sequence databases but the feasibility of this approach was only demonstrated on a small subset of MS/MS data (doubly-charged peptides 10-14 amino acids long).

Our novel ranking-based score helps ameliorate the two main deficiencies of proteogenomic mapping: speed and sensitivity. Using PepNovo-generated tags we are able to perform the database search significantly faster than the current state-of-the-art (approximately 15 times faster than InsPecT). In addition, reranking the results using our new scoring function significantly increases the number of identified peptides compared to the results obtained by a regular run of InsPecT.

We performed a benchmark experiment with the same HEK293 run used above to search the IPI sequence database. This time our sequence database was a six-frame translation of the human genome (NCBI build 35.3 masked using RepeatMasker), which contained approximately 3 billion amino acids - one hundred times larger than the IPI sequence database. The sequences in the six-frame translation were split into 40 files, and each sequence file was searched separately. The results of the 40 searches were then pooled and filtered, keeping the ten highest scoring peptide identifications for each spectrum. Similarly to the experiments with the IPI database, we ran three different types of searches: “Regular InsPecT”, “InsPecT Tags + Rank Score”, and “PepNovo Tags + Rank Score”.

Table 4 reports the results of these benchmark searches against a six-frame translation of the human genome. Like our experiments with the IPI database, we see a significant improvement when using PepNovo's tags and the new ranking-based score. However, the advantages of our new scoring become much more significant with the six-frame translation's challenging one hundredfold larger search space. There is a 61.3% increase in the number of identified peptides with “PepNovo Tags + Rank Score” search compared to a regular InsPecT run, and a 38.9% increase when only the reranking of results is applied. This increase is almost three times larger than the increase observed when we searched the IPI sequence database.

The total number of peptides identified in the six-frame translation is significantly lower than the number identified when searching IPI. However, while the regular InsPecT search losses 55% of its peptide identifications (it goes from 22518 peptides identifications down to 10356), the “PepNovo Tags + Rank Score” method fares significantly better, losing only 40% of its identifications (from 27685 peptides down to 16706). Similar reductions in the number of peptide identifications have been observed with previous proteogenomic experiments.[74] One reason behind these reductions in identifications is the limited discriminatory power of the scoring functions. With a six-frame translation we encounter many more high-scoring incorrect PSMs compared to the number encountered when searching a significantly smaller search space. This reduces the number of positive identifications that can be accepted at a given false positive rate. For example, a 30% reduction was observed in the number of identified peptides that fell within exonic regions when switching from a protein sequence database to a six-frame translation of the genome of *Arabidopsis thaliana*.[72] In addition, many of the peptides identified when searching protein sequence databases happen to fall on exonic boundaries. These products of splice events are not present in six-frame translations, and are bound to be missed. The number of such cases can be surprisingly large. In one case, 36.4% of the identifications made when searching MS/MS spectra against the human IPI sequence database belonged to peptides that spanned exonic boundaries.[61]

Table 5 compares between the peptide identifications made searching against IPI with the identifications made searching against the six-frame translation of the human genome. As expected, by comparing the second and third columns in the table, we see that the number of identifications made in the six-frame search is much smaller than the number of identifications made in the search against the one hundred times smaller IPI protein database. In fact, the fourth column shows that with all three search methods, a little more than a third of the peptides identified in the IPI search are lost because they do not appear in the six-frame translation (they span exonic boundaries). Many peptides that were identified in IPI and were also present in the six-frame translation, were not included in the final set of positive identifications form the six-frame search. The culprit in this case was the larger search space, which greatly increased the number of high-scoring incorrect PSMs. These additional high-scoring incorrect PSMs raise the scoring bar that needs to be passed for a PSM to be accepted as positive identification. Thus, these lost identifications can be directly attributed to deficiencies in the scoring functions. From this perspective, our novel ranking score performs significantly better than the default InsPecT scoring function. The fifth column shows that while a regular InsPecT run lost 5103 of the 22518 peptides it identified in IPI (22.7%), rescoring the InsPecT results using our novel score reduced this loss to 3037 from 25827 (11.8%). Interestingly, the search that used PepNovo tags lost only 2449 of the 27685 identified peptides (8.8%). The reason PepNovo's tags lose fewer peptide identifications is that these tags are 100 times more efficient than the tags used by InsPecT. Their higher filtration rate results in fewer high-scoring incorrect PSMs, which ultimately lowers the score threshold required to accept positive matches.

The last column in Table 5 lists the number of peptide identifications that are unique to the six-frame translation, representing products of unannotated genes. All three search methods show a similar ratio of these peptide identifications (between 5.7% and 6.9%). However since many more peptides got identified with “PepNovo Tags + Rank Score”, the number of novel peptides found with this method (1153), is significantly higher than the number found using a regular InsPecT search (625) or a rescored InsPecT run (828).

Discussion

In this paper we explored how discriminative data-driven ranking models could be used successfully for the complex task of scoring peptide-spectrum matches (PSMs). We argue that this scoring problem is inherently a ranking problem, especially for de novo sequencing where the goal is simply to bring the correct PSM for spectrum ahead of all the incorrect ones (for the same spectrum). This led us to solve this problem with a machine learning ranking algorithm (RankBoost30), rather than use more common classification-oriented generative approaches.

When creating our scoring models, we had at our disposal a large set of diverse features which described many aspects that are known to be indicative of a poor or strong PSM. These features examined various attributes like the peptide's path in the spectrum graph, peak annotations (e.g., the numbers of *b*,*y*-ions that got annotated), the peptide's sequence (characteristics of proteotypic peptides), and more. An important source of features were based on predictions of peak ranks, that were done using a novel ranking-based algorithm we developed.[31] Each of these features, in its own right, might be only marginally successful at discriminating between a correct and incorrect PSM; constituting what is called a “weak learner”. The RankBoost algorithm[30] proved to be a very effective at combining this diverse set of features into powerful and discriminatory scoring function. In addition, by examining the models created by RankBoost we were able to gain insight into the dynamics and contributions of the various features (see supplemental material).

We demonstrated how our novel scoring function can be used to deliver superior performance in several MS/MS scoring tasks. By reranking the original order of the de novo results we were able significantly improved PepNovo's accuracy rates. This boosted the algorithm's performance well above the current state-of-the-art. For instance, when making a single sequence prediction, our reranked results are 10%-20% higher than the current high-performance algorithms (PepNovo and the commercial software Peaks). This performance gap persists even for larger sets of predictions (see Figure 3 and Figure 4).

Our novel score also greatly enhanced the accuracy of PepNovo's peptide sequence tags (see supplemental material). This enabled us to generate longer tags without compromising their ability to be a covering set (i.e., a set which contains at least one completely correct tag). The enhanced tagging capability both increased the accuracy of database searches and significantly reduced the running time, enabling us to speedup InsPecT's searches against a six-frame translation of the human genome by a factor of 15.

In addition to developing scoring models for de Novo sequencing, we trained specific models for rescoring database search results. When used to rescore results of InsPecT runs that searched MS/MS spectra against the human IPI protein sequences (~ 30 million amino acids), our new score was able to increase the number of peptide identifications by 14.7%. The increase grew to 22.9% when the search used PepNovo's tags instead of the default ones generated by InsPecT. However, the benefits of our novel scoring method were more substantial when applied to results of a search against a six-frame translation of the human genome (~ 3 billion amino acids). When we rescored InsPecT's results we witnessed a 38.9% increase in the number of identified peptides; using PepNovo tags along with the ranking-based scoring led to a higher increase of 61.3%. With our novel scoring method we also almost doubled the number of novel peptide identifications belonging to unannotated genes (increasing the 625 new peptides found in a regular InsPecT search to 1153).

Our experimental results underscore the fact that our models perform particularly well in challenging domains that have large search spaces. This trait becomes especially important when we start to consider more and more complex analysis tasks, such as searches that consider alternative splicing,[32] large-scale “blind” searches,[33,76] and even searches for fusion proteins.[77] The search spaces in these domains can be so large, and contain so many high-scoring incorrect PSMs, that without more powerful scoring functions, it will be impossible to accept but a handful of the highest-scoring, and most obvious, identifications. Many interesting identification will be lost since they will lack statistical significance with current scoring methods. Our scoring function, which can be used as a stand-alone post-processing operation, can help increase the number of interesting discoveries made in such experiments.

Synopsis

We present novel approach to scoring of peptide-spectrum matches. We use discriminative boosting-based algorithms that are able to harnesses the large volume of MS/MS data presently available to train more accurate scoring models. Our new method improves the performance of our de novo sequencing algorithm beyond the current state-of-the-art, and also greatly enhances the performance of database search programs.

Supplementary Material

Refer to Web version on PubMed Central for supplementary material.

Acknowledgement

I would like to thank Natalie Castellana and Samuel Payne for helpful discussions and their assistance in running benchmark experiments. I am grateful to Pavel Pevzner form many inspiring discussions on the various topics covered by this manuscript. This research was supported by the “National Center for Research Resources” of the NIH via grant

P-41-RR24851. I would also like to acknowledge the UCSD FWGrid Project for the availability of their computational infrastructure. The UCSD FWGrid Project is funded in part by NSF Research Infrastructure Grant number NSF EIA-0303622.

References

(1). Auerbach D, Thaminy S, Hottiger M, Stagljar I. The post-genomic era of interactive proteomics: facts and perspectives. Proteomics 2002;2:611–23. [PubMed: 12112840]

(2). Pandey A, Mann M. Proteomics to study genes and genomes. Nature 2000;405:837–846. [PubMed: 10866210]

(3). Washburn M, Wolters D, Yates J III. Large Scale analysis of the yeast proteome via multidimensional protein identification technology. Nature Biotech 2001;19:242–247.

(4). Aebersold R, Mann M. Mass spectrometry-based proteomics. Nature 2003;422:198–207. [PubMed: 12634793]

(5). Stein S, Scott D. Optimization and Testing of Mass Spectral Library Search Algorithms for Compound Identification. J. Am. Soc. Mass. Spectrom 1994;5:859–866.

(6). Yates J III, Morgan S, Gatlin P, C.L. Griffin, Eng J. Method To Compare Collision-Induced Dissociation Spectra of Peptides: Potential for Library Searching and Subtractive Analysis. Anal. Chem 1998;70:3557–3565. [PubMed: 9737207]

(7). Eng J, McCormack A, Yates J III. An Approach to Correlate Tandem Mass-Spectral Data of Peptides with Amino Acid Sequences in a Protein Database. J. Am. Soc. Mass Spectrom 1994;5:976–989.

(8). Dancík V, Addona T, Clauser K, Vath J, Pevzner P. De novo peptide sequencing via tandem mass spectrometry. J. Comput. Biol 1999;6:327–342. [PubMed: 10582570]

(9). Perkins D, Pappin D, Creasy D, Cottrell J. Probability-based protein identification by searching sequence databases using mass spectrometry data. Electrophoresis 1999;20:3551–3567. [PubMed: 10612281]

(10). Bafna V, Edwards N. SCOPE: a probabilistic model for scoring tandem mass spectra against a peptide database. Bioinformatics 2001;17(Suppl 1):13–21. [PubMed: 11222258]

(11). Colinge J, Masselot A, Giron M, Dessingy T, Magnin J. OLAV: Towards high-throughput tandem mass spectrometry data identification. Proteomics 2003;3:1454–1463. [PubMed: 12923771]

(12). Ma B, Zhang K, Hendrie C, Liang C, Li M, Doherty-Kirby A, Lajoie G. PEAKS: powerful software for peptide de novo sequencing by tandem mass spectrometry. Rapid Commun. Mass Spectrom 2003;17:2337–2342. [PubMed: 14558135]

(13). Havilio M, Haddad Y, Smilansky Z. Intensity-based statistical scorer for tandem mass spectrometry. Anal. Chem 2003;75:435–444. [PubMed: 12585468]

(14). Colinge J, Masselot A, Cusin I, Mahé E, Niknejad A, Argoud-Puy G, Reffas S, Bederr N, Gleizes A, Rey P, Bougueleret L. High-performance peptide identification by tandem mass spectrometry allows reliable automatic data processing in proteomics. Proteomics 2004;4:1977–1984. [PubMed: 15221758]

(15). Elias J, Gibbons F, King O, Roth F, Gygi S. Intensity-based protein identification by machine learning from a library of tandem mass spectra. Nat. biotech 2004;22:214–219.

(16). Frank A, Pevzner P. PepNovo: De Novo Peptide Sequencing via Probabilistic Network Modeling. Anal. Chem 2005;77:964–973. [PubMed: 15858974]

(17). Tanner S, Shu H, Frank A, Mumby M, Pevzner P, Bafna V. InsPecT: Fast and accurate identification of post-translationally modified peptides from tandem mass spectra. Anal. Chem 2005;77:4626–4639. [PubMed: 16013882]

(18). Fischer B, Roth V, Roos F, Grossmann J, Baginsky S, Widmayer P, Gruissem W, Buhmann J. NovoHMM: A Hidden Markov Model for de Novo Peptide Sequencing. Anal. Chem 2005;77:7265–7273. [PubMed: 16285674]

(19). Cannon W, Jarman K, Webb-Robertson B-J, Baxter D, Oehmen C, Jarman K, Heredia-Langner A, Auberry K, Anderson G. Comparison of Probability and Likelihood Models for Peptide Identification from Tandem Mass Spectrometry Data. J. of Proteome Res 2005;4:1687–1698. [PubMed: 16212422]

(20). Wan Y, Chen T. PepHMM: A hidden Markov model based scoring function for tandem mass spectrometry. Anal. Chem 2006;78:432–7. [PubMed: 16408924]

(21). Sadygov R, Wohlschlegel J, Park S, Xu T, Yates J III. Central limit theorem as an approximation for intensity-based scoring function. Anal. Chem 2006;78:89–95. [PubMed: 16383314]

(22). Colinge J. Peptide Fragment Intensity Statistical Modeling. Anal. Chem 2007;79:7286–7290. [PubMed: 17713966]

(23). Mo L, Dutta D, Wan Y, Chen T. MSNovo: A Dynamic Programming Algorithm for de Novo Peptide Sequencing via Tandem Mass Spectrometry. Anal. Chem 2007;79:4870–4878. [PubMed: 17550227]

(24). Bern M, Cai Y, Goldberg D. Lookup Peaks: A Hybrid of de Novo Sequencing and Database Search for Protein Identification by Tandem Mass Spectrometry. Anal. Chem 2007;79:1393–1400. [PubMed: 17243770]

(25). Klammer A, Reynolds S, Bilmes J, MacCoss M, Noble W. Modeling peptide fragmentation with dynamic Bayesian networks for peptide identification. Bioinformatics 2008;24:i348–356. [PubMed: 18586734]

(26). Frank A, Savitski M, Nielsen M, Zubarev R, Pevzner P. De Novo Peptide Sequencing and Identification with Precision Mass Spectrometry. J. Proteome Res 2007;6:114–123. [PubMed: 17203955]

(27). Kim S, Gupta N, Pevzner P. Spectral Probabilities and Generating Functions of Tandem Mass Spectra: A Strike against Decoy Databases. J. Proteome Res 2008;7:3354–3363. [PubMed: 18597511]

(28). Keller A, Nesvizhskii A, Kolker E, Aebersold R. Empirical statistical model to estimate the accuracy of peptide identifications made by MS/MS and database search. Anal. Chem 2002;74:5383–5392. [PubMed: 12403597]

(29). Käll L, Canterbury J, Weston J, Noble W, MacCoss M. Semi-supervised learning for peptide identification from shotgun proteomics datasets. Nat. Methods 2007;4:923–25. [PubMed: 17952086]

(30). Freund Y, Iyer R, Schapire R, Singer Y. An Efficient Boosting Algorithm for Combining Preferences. Journal of Machine Learning Research 2003;4:933–969.

(31). Frank, A. submitted

(32). Tanner S, Shen Z, Ng J, Florea L, Guigó R, Briggs S, Bafna V. Improving gene annotation using peptide mass spectrometry. Genome Res 2007;17:231–239. [PubMed: 17189379]

(33). Tanner S, Payne SH, Dasari S, Shen Z, Wilmarth PA, David LL, Loomis WF, Briggs SP, Bafna V. Accurate Annotation of Peptide Modifications through Unrestrictive Database Search. J. Proteome Res 2008;7:170–181. [PubMed: 18034453]

(34). Masselon C, Pasa-Tolic L, Tolic N, Anderson G, Bogdanov B, Vilkov A, Shen Y, Zhao R, Qian W, Lipton M, Camp D, Smith R. Targeted Comparative Proteomics by Liquid Chromatography-Tandem Fourier Ion Cyclotron Resonance Mass Spectrometry. Anal. Chem 2005;77:400–406. [PubMed: 15649034]

(35). Gupta N, Tanner S, Jaitly N, Adkins J, Lipton M, Edwards R, Romine M, Osterman A, Bafna V, Smith R, Pevzner P. Whole proteome analysis of post-translational modifications: applications of mass-spectrometry for proteogenomic annotation. Genome Res 2007;17:1362–1377. [PubMed: 17690205]

(36). Higdon R, Hogan J, Belle GV, Kolker E. Randomized sequence databases for tandem mass spectrometry peptide and protein identification. OMICS 2005;9:364–379. [PubMed: 16402894]

(37). Elias J, Gygi S. Target-decoy search strategy for increased confidence in large-scale protein identifications by mass spectrometry. Nat. Methods 2007;4:207–214. [PubMed: 17327847]

(38). Rosenblatt F. The Perceptron: A Probabilistic Model for Information Storage and Organization in the Brain. Psychological Review 1958;65:386–408. [PubMed: 13602029]

(39). Duda, R.; Hart, P. Pattern Classification and Scene Analysis. Wiley-Interscience; 1973.

(40). Vapnik, V. The Nature of Statistical Learning Theory. Springer; New York, NY, USA: 1995.

(41). Freund Y, Schapire R. A decision-theoretic generalization of on-line learning and an application to boosting. Journal of Computer and System Sciences 1997;55:119–139.

(42). Schapire R, Singer Y. Improved Boosting Using Confidence-rated Predictions. Machine Learning 1999;37:297–336.

(43). Wysocki V, Tsaprailis G, Smith L, Breci L. Mobile and Localized Protons: A Framework for Understanding Peptide Dissociation. J. Mass Spectrom 2000;35:1399–1406. [PubMed: 11180630]

(44). Tabb D, Smith L, Breci L, Wysocki V, Lin D, Yates J III. Statistical characterization of ion trap tandem mass spectra from doubly charged tryptic peptides. Anal. Chem 2003;75:1155–1163. [PubMed: 12641236]

(45). Sun S, Meyer-Arendt K, Eichelberger B, Brown R, Yen C-Y, Old W, Pierce K, Cios K, Ahn N, Resing K. Improved Validation of Peptide MS/MS Assignments Using Spectral Intensity Prediction. Mol. Cell. Proteomics 2007;6:1–17. [PubMed: 17018520]

(46). Zhang Z. Prediction of Low-Energy Collision-Induced Dissociation Spectra of Peptides. Anal. Chem 2004;76:3908–3922. [PubMed: 15253624]

(47). Bartels C. Fast algorithm for peptide sequencing by mass spectroscopy. Biomedical and Environmental Mass Spectrometry 1990;19:363–8.

(48). Craig R, Cortens J, Beavis R. The use of proteotypic peptide libraries for protein identification. Rapid Commun. Mass Spectrom 2005;19:1844–1850. [PubMed: 15945033]

(49). Tang H, Arnold R, Alves P, Xun Z, Clemmer D, Novotny M, Reilly J, Radivojac P. A computational approach toward label-free protein quantification using predicted peptide detectability. Bioinformatics 2006;22:e481–488. [PubMed: 16873510]

(50). Mallick P, Schirle M, Chen S, Flory M, Lee H, Martin D, Ranish J, Raught B, Schmitt R, Werner T, B. K, Aebersold R. Computational prediction of proteotypic peptides for quantitative proteomics. Nature Biotech 2007;25:125–131.

(51). Mann M, Wilm M. Error-Tolerant Identification of Peptides in Sequence Databases by Peptide Sequence Tags. Anal. Chem 1994;66:4390–4399. [PubMed: 7847635]

(52). Sunyaev S, Liska A, Golod A, Shevchenko A, Shevchenko A. MultiTag: multiple error-tolerant sequence tag search for the sequence-similarity identification of proteins by mass spectrometry. Anal. Chem 2003;75:1307–1315. [PubMed: 12659190]

(53). Tabb D, Saraf A, Yates J III. GutenTag: High-throughput sequence tagging via an empirically derived fragmentation model. Anal. Chem 2003;75:6415–6421. [PubMed: 14640709]

(54). Frank A, Tanner S, Bafna V, Pevzner P. Peptide sequence tags for fast database search in mass-spectrometry. J. Proteome Res 2005;4:1287–95. [PubMed: 16083278]

(55). Shilov I, Seymour S, Patel A, Loboda A, Tang W, Keating S, Hunter C, Nuwaysir, L, Schaeffer D. The Paragon Algorithm, a Next Generation Search Engine That Uses Sequence Temperature Values and Feature Probabilities to Identify Peptides from Tandem Mass Spectra. Mol. Cell. Proteomics 2007;6:1638–1655. [PubMed: 17533153]

(56). Shevchenko A, Loboda A, Sunyaev S, Shevchenko A, Bork P, Ens W, Standing K. Charting the proteomes of organisms with unsequenced genomes by MALDI-Quadrupole Time-of Flight Mass Spectrometry and BLAST homology searching. Anal. Chem 2001;73:1917–1926. [PubMed: 11354471]

(57). Searle B, Dasari S, Turner M, Reddy A, Choi D, Wilmarth P, McCormack A, David L, Nagalla S. High-throughput identification of proteins and unanticipated sequence modifications using a mass-based alignment algorithm for MS/MS de novo sequencing results. Anal. Chem 2004;76:2220–2230. [PubMed: 15080731]

(58). Han Y, Ma B, Zhang K. SPIDER: software for protein identification from sequence tags with de novo sequencing error. J Bioinform. Comput. Biol 2005;3:697–716. [PubMed: 16108090]

(59). Keller A, Purvine S, Nesvizhskii A, Stolyar S, Goodlett D, Kolker E. Experimental protein mixture for validating tandem mass spectral analysis. OMICS 2002;6:207–212. [PubMed: 12143966]

(60). Pevtsov S, Fedulova I, Mirzaei H, Buck C, Zhang X. Performance Evaluation of Existing De Novo Sequencing Algorithms. J. Prot. Research 2006;5:3018–3028.

(61). Kim S, Gupta N, Bandeira N, Pevzner P. Spectral dictionaries: Integrating de novo peptide sequencing with database search of tandem mass spectra. Mol. Cell. Proteomics 2009;8:53–69. [PubMed: 18703573]

(62). Klimek J, Eddes J, Hohmann L, Jackson J, Peterson A, Letarte S, Gafken P, Katz J, Mallick P, Lee H, Schmidt A, Ossola R, Eng J, Aebersold R, Martin D. The Standard Protein Mix Database: A

Diverse Data Set To Assist in the Production of Improved Peptide and Protein Identification Software Tools. J. Proteome Res 2008;7:96–103. [PubMed: 17711323]

(63). Siepel A, et al. Targeted discovery of novel human exons by comparative genomics. Genome Res 2007;17:1763–73. [PubMed: 17989246]

(64). Stark A, et al. Discovery of functional elements in 12 Drosophila genomes using evolutionary signatures. Nature 2007;450:219–232. [PubMed: 17994088]

(65). Choudhary J, Blackstock W, Creasy D, Cottrell J. Matching peptide mass spectra to EST and genomic DNA databases. Trends Biotechnol 2001;19:S17–S22. [PubMed: 11780965]

(66). Jaffe J, Berg H, Church G. Proteogenomic mapping as a complementary method to perform genome annotation. Proteomics 2004;4:59–77. [PubMed: 14730672]

(67). Desiere F, et al. Integration with the human genome of peptide sequences obtained by high-throughput mass spectrometry. Genome Biol 2005;6:R9. [PubMed: 15642101]

(68). Kalume D, Peri S, Reddy R, Zhong J, Okulate M, Kumar N, Pandey A. Genome annotation of Anopheles gambiae using mass spectrometry-derived data. BMC Genomics 2005;6:128. [PubMed: 16171517]

(69). Fermin D, Allen B, Blackwell T, Menon R, Adamski M, Xu Y, Ulintz P, Omenn G, States D. Novel gene and gene model detection using a whole genome open reading frame analysis in proteomics. Genome Biol 2006;7:R35. [PubMed: 16646984]

(70). Ansong C, Purvine S, Adkins J, Lipton M, Smith R. Proteogenomics: needs and roles to be filled by proteomics in genome annotation. Brief Funct Genomic Proteomic 2008;7:50–62. [PubMed: 18334489]

(71). Sevinsky J, Cargile B, Bunger M, Meng F, Yates N, Hendrickson R, Stephenson J Jr. Whole Genome Searching with Shotgun Proteomic Data: Applications for Genome Annotation. J. Proteome Res 2008;7:80–88. [PubMed: 18062665]

(72). Castellana N, Payne S, Shen Z, Stanke M, Briggs S, Bafna V. Discovery and revision of Arabidopsis genes by proteogenomics. PNAS 2008;105:21034–21038. [PubMed: 19098097]

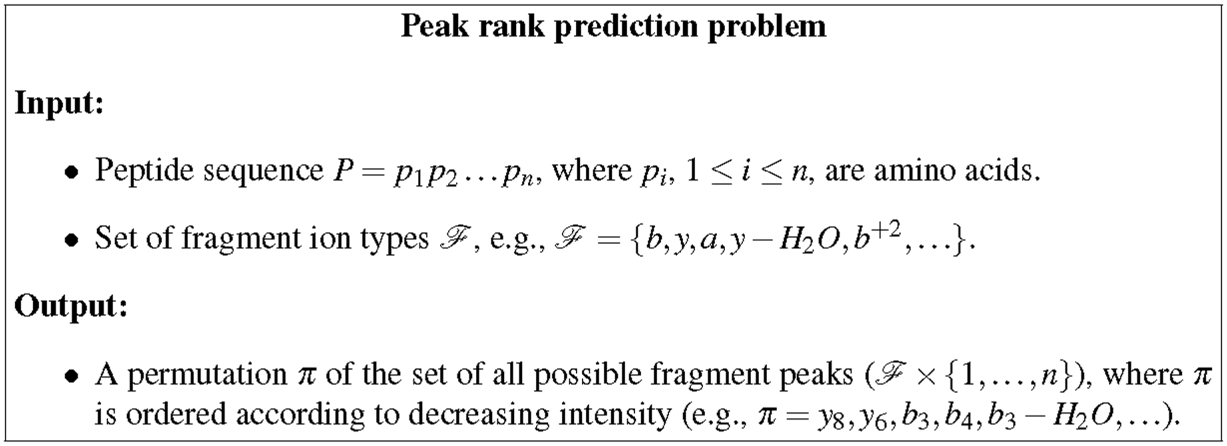
(73). Payne S, Yau M, Smolka M, Tanner S, Zhou H, Bafna V. Phosphorylation specific MS/MS scoring for rapid and accurate phospho-proteome analysis. J. Proteome Res 2008;7:3373Ű–3381. [PubMed: 18563926]

(74). Colinge J, Cusin I, Reffas S, Mahe E, Niknejad A, Rey P-A, Mattou H, Moniatte M, Bougueleret L. Experiments in Searching Small Proteins in Unannotated Large Eukaryotic Genomes. J. Proteome Res 2005;4:167–174. [PubMed: 15707372]

(75). Craig R, Beavis R. TANDEM: matching proteins with tandem mass spectra. Bioinformatics 2004;20:1466–1467. [PubMed: 14976030]

(76). Tsur D, Tanner S, Zandi E, Bafna V, Pevzner P. Identification of Post-translational Modifications via Blind Search of Mass-Spectra. Nature Biotech 2005;23:1562–2567.

(77). Ng J, Pevzner P. Algorithm for Identification of Fusion Proteins via Mass Spectrometry. J. Proteome Res 2008;7:89–95. [PubMed: 18173219]



Peak rank prediction problem

Input:

- Peptide sequence $P = p_1 p_2 \ldots p_n$, where p_i, $1 \leq i \leq n$, are amino acids.
- Set of fragment ion types $\mathscr{F}$, e.g., $\mathscr{F} = \{b, y, a, y - H_2O, b^{+2}, \ldots\}$.

Output:

- A permutation π of the set of all possible fragment peaks $(\mathscr{F} \times \{1, \ldots, n\})$, where π is ordered according to decreasing intensity (e.g., $\pi = y_8, y_6, b_3, b_4, b_3 - H_2O, \ldots$).

Figure 1.

The peak rank prediction problem.

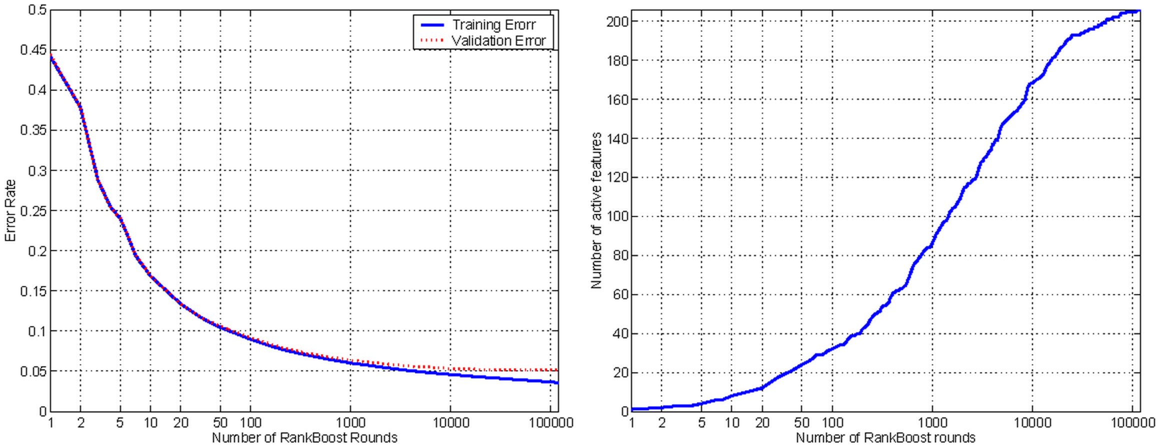


Figure 2.

Training of a de novo PSM scoring model. The graph on the left displays the training and validation error rates after running the RankBoost algorithm for various numbers of rounds. The graph on the right displays the number of active features in the model (i.e., features that have a nonzero wight). The *x*-axis displays the number of boosting rounds using a logarithmic scale. The figures were generated for a training set of doubly-charged peptides of mass 1100-1300.

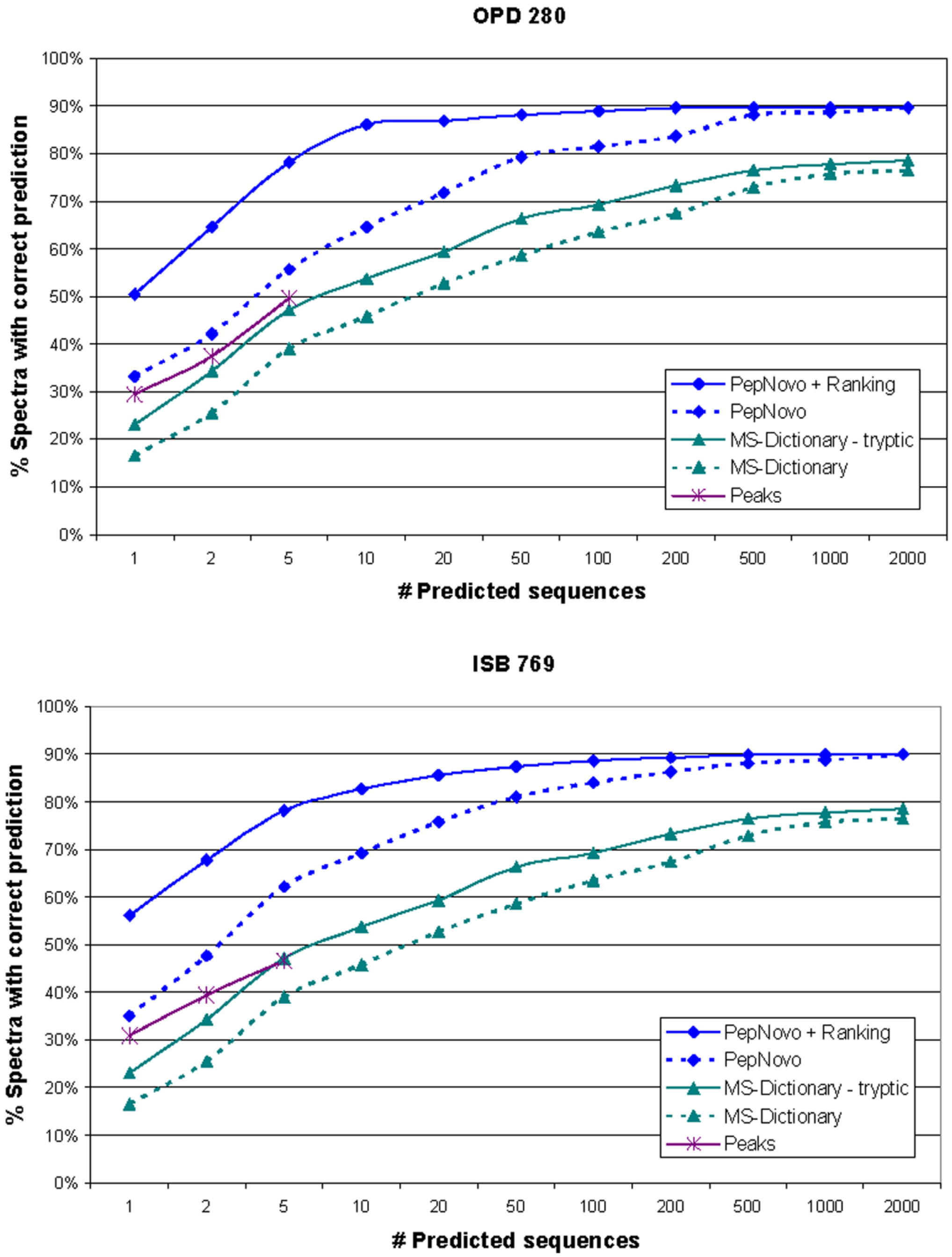


Figure 3.

Benchmark results for OPD280 and ISB769. The plots show results for PepNovo (with and without reranking), MS-Dictionary (with tryptic only and non-restricted predictions), and Peaks. In each plot the *x*-axis shows the size of the set of highest scoring predicted sequences (1-2000), and the *y*-axis shows the proportion of spectra for which the set of de novo predictions contained a correct sequence.

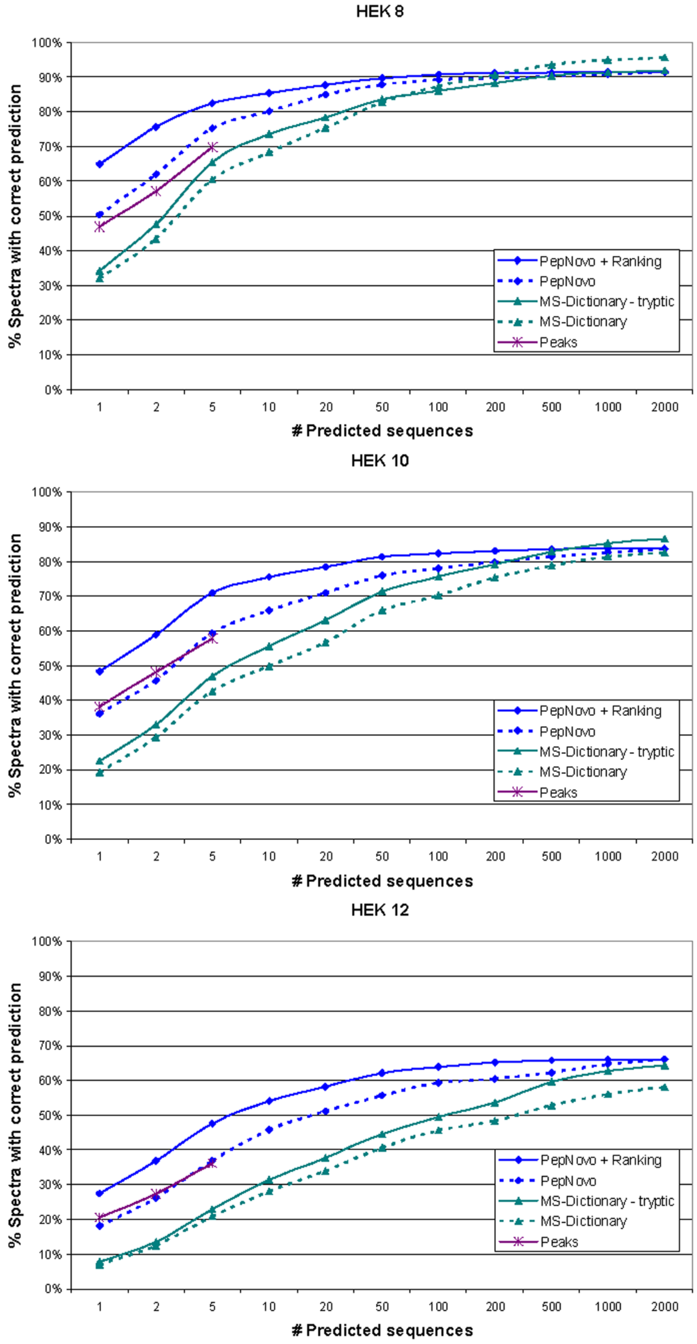


Figure 4.

Benchmark results for sets HEK8,HEK10 and HEK12. The plots show results for Pep-Novo (with and without reranking), MS-Dictionary (with tryptic only and non-restricted predictions), and Peaks. In each plot the *x*-axis shows the size of the set of highest scoring predicted sequences (1-2000), and the *y*-axis shows the proportion of spectra for which the set of de novo predictions contained a correct sequence.

Table 1

MS/MS training dataset. The set of 319578 pairs of unique peptides and MS/MS spectrawas partitioned according to charge and precursor mass. For each partition we list its precursor mass range, the number of peptides that fell in that range, and the typical length of those peptides (the lengths listed cover at least 95% of the peptides in each partition).

Charge 1			Charge 2			Charge 3		
Precursor mass (Da)	**#Unique peptides**	**Typical lengths**	**Precursor mass (Da)**	**#Unique peptides**	**Typical lengths**	**Precursor mass (Da)**	**#Unique peptides**	**Typical lengths**
0-1150	20971	7-12	0-1100	25709	7-12	0-1950	13198	10-19
1150-1400	18984	9-15	1100-1300	33167	9-14	1950-2450	13131	16-24
1400+	16231	11-20	1300-1600	45595	10-16	2450-3000	12684	20-29
			1600-1900	43054	13-20	3000+	13824	25-48
			1900-2400	43225	15-25			
			2400+	19805	20-32			
56186			210555			52837		

Table 2

Average prediction lengths in de novo benchmarking experiments. For each dataset we note the average length of the top-ranked correct predictions.

	Average Predicted Length	
Algorithm	**OPD280 (10.5)***	**ISB769 (11.7)**[a]
PepNovo	10.2	10.7
PepNovo + Ranking	8.6	9.3
MS-Dictionary - tryptic	10.4	11.7
MS-Dictionary	10.5	11.8
Peaks	10.2	11.4

*The average length of the peptides in the OPD280 dataset was 10.5 amino acids, and in the ISB769 dataset it was 11.7 amino acids.

Table 3

Database search results for a HEK293 run of 750000 spectra against the IPI sequence database (version 3.42). The table compares the results obtained by using Inspect in the default mode ("Regular Inspect"), rescoring Inspect results ("Inspect Tags + Rank Score"), and using PepNovo tags and rescoring results ("PepNovo Tags + Rank Score"). The identifications were made with a false discovery rate of 1% at the spectrum level which is approximately 4% at the peptide level. The values in parentheses indicate the relative gain in identifications compared to the regular InsPecT search.

	Search type		
Identifications	**Regular InsPecT**	**InsPecT Tags + Rank Score**	**PepNovo Tags + Rank Score**
Spectra:			
Charge 1	6891	10017 (+45.3%)	13134 (+90.5%)
Charge 2	89259	96244 (+7.8%)	99775 (+11.8%)
Charge 3	14284	19516 (+36.6%)	18324 (+28.3%)
Total	110434	125577 (+13.7%)	131233 (+18.8%)
Pep tides:			
Charge 1	3961	5721 (+44.4%)	6977 (+76.1%)
Charge 2	20304	22061 (+8.7%)	23526 (+15.9%)
Charge 3	3217	4586 (+42.5%)	4450 (+38.3%)
Total (unique)	22518	25827 (+14.7%)	27685 (+22.9%)

Table 4

Database search results for a HEK293 run with 750000 spectra against a six-frame translation of the human genome (NCBI build 35.3 masked using RepeatMasker). The table compares the results obtained by using Inspect in the default mode ("Regular Inspect"), rescoring Inspect results ("Inspect Tags + Rank Score"), and using PepNovo tags and rescoring results ("PepNovo Tags + Rank Score"). The identifications were made with a false discovery rate of 1% at the spectrum level which is approximately 4% at the peptide level. The values in parentheses indicate the relative gain in identifications compared to the regular InsPecT search.

	Search type		
Identifications	**Regular InsPecT**	**InsPecT Tags + Rank Score**	**PepNovo Tags + Rank Score**
Spectra:			
Charge 1	3109	5836 (+87.7%)	7268 (+133.7%)
Charge 2	39997	53107 (+32.7%)	61855 (+54.6%)
Charge 3	4529	9557 (+111.0%)	10426 (+130.2%)
Total	47635	68500 (+43.8%)	79549 (+66.9%)
Peptides:			
Charge 1	1761	3279 (+86.2%)	3820 (+116.9%)
Charge 2	9430	12326 (+30.7%)	13725 (+45.5%)
Charge 3	1020	2244 (+120.0%)	2945 (+188.7%)
Total (unique)	10356	14391 (+38.9%)	16706 (+61.3%)

Table 5

Comparison between identifications made with IPI and six-frame searches. The table compares the results obtained by using InsPecT in the default mode ("Regular Inspect"), rescoring Inspect results ("InsPecT Tags + Rank Score"), and using PepNovo tags along with rescoring of the results ("PepNovo Tags + Rank Score").

Search Type	**# Peptides identified when searching against IPI**	**# Peptides identified when searching six-frame translation**	**# Peptides identified in IPI that were not in the six-frame translation DB**	**# Peptides from IPI that were in six-frame DB, but lost due to deficient scoring**	**# Novel peptides identified only in six-frame search**
Regular InsPecT	22518	10356	7684	5103	625
InsPecT Tags + Rank Score	25827	14391	9227	3037	828
PepNovo Tags + Rank Score	27685	16706	9683	2449	1153